

Задача А. Разрез

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Найдите минимальный разрез между вершинами 1 и n в заданном неориентированном графе.

Формат входных данных

На первой строке входного файла содержится n ($1 \leq n \leq 100$) — число вершин в графе и m ($0 \leq m \leq 400$) — количество ребер. На следующих m строках входного файла содержится описание ребер. Ребро описывается номерами вершин, которые оно соединяет, и его пропускной способностью (положительное целое число, не превосходящее 10 000 000), при этом никакие две вершины не соединяются более чем одним ребром.

Формат выходных данных

На первой строке выходного файла должны содержаться количество ребер в минимальном разрезе и их суммарная пропускная способность. На следующей строке выведите возрастающую последовательность номеров ребер (ребра нумеруются в том порядке, в каком они были заданы во входном файле).

Пример

стандартный ввод	стандартный вывод
3 3 1 2 3 1 3 5 3 2 7	2 8 1 2

Задача В. Милые панды

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

В городе живут n панд, пронумерованные числами от 1 до n , у i -й из которых a_i пончиков. Также есть n ящиков, пронумерованных от 1 до n , в i -м из которых помещается b_i пончиков. Для каждого i от 1 до n , i -я панда может распределить свои пончики между i -м и $(i \bmod n + 1)$ -м ящиками, при этом некоторые пончики может оставить себе.

Найдите максимальное возможное суммарное количество пончиков во всех ящиках.

Формат входных данных

Входной файл содержит 0 или более тестов. Описание одного теста:

Первая строка содержит целое число n ($3 \leq n \leq 10^6$).

Вторая строка содержит n целых чисел a_1, a_2, \dots, a_n ($0 \leq a_i \leq 10^9$).

На третьей строке n целых чисел b_1, b_2, \dots, b_n ($0 \leq b_i \leq 10^9$).

Гарантируется, что сумма n по всем тестам не превышает 10^6 .

Формат выходных данных

Для каждого теста в отдельной строке выведите одно целое число — ответ на задачу.

Пример

стандартный ввод	стандартный вывод
5	11
8 4 8 3 10	13
1 0 4 5 1	
5	
9 4 10 0 4	
3 5 2 2 1	

Задача С. Ремесленник

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Такеши, будучи известным ремесленником, принимает много заказов со всей Японии. Однако, инструменты, которые он использует, сильно устарели, поэтому он хочет купить новые и заменить ими устаревшие перед следующим использованием.

Для выполнения некоторых заказов ему придется купить новые инструменты. Таким образом, ему не всегда выгодно выполнить все заказы. Вы являетесь учеником Такеши. Вашим заданием является найти набор заказов, которые ему нужно выполнить, чтобы чистая прибыль была максимальной. Чистая прибыль зависит только от дохода от заказа и потраченной им суммы на замену инструментов.

Он покупает инструменты только у своих друзей. Некоторые друзья предлагают скидку для некоторых пар инструментов, если купить их одновременно. Не забудьте учитывать скидки. Суммарная стоимость не всегда равна сумме цен купленных инструментов. Все приобретенные инструменты достаточно прочные, чтобы их хватило на выполнение всех заказов: достаточно купить не более чем 1 штуки каждого вида.

Формат входных данных

Ввод удовлетворяет следующему формату.

$N \ M \ P$

$X_1 K_1 I_{1,1} \cdots I_{1,K_1}$

\cdots

$X_N K_N I_{N,1} \cdots I_{N,K_N}$

Y_1

\cdots

Y_M

$J_{1,1} J_{1,2} D_1$

\cdots

$J_{P,1} J_{P,2} D_P$

, где N , M , P - количество заказов, продаваемых инструментов и пар, которые можно купить по скидке соответственно.

Следующие N строк описывают заказы. X_i плата, получаемая за i -ый заказ, и K_i количество инструментов, нужных для выполнения заказа.

Затем K_i целых чисел, описывающие инструменты. Все инструменты определяются числом от 1 до M .

Следующие M строк содержат цены инструментов у друзей Такеши. Целое число Y_i представляет собой цену i -ого инструмента.

Последние P строк описывают пары, продаваемые по скидке. Если Такеши покупает $J_{i,1}$ -ый и $J_{i,2}$ -ый инструменты одновременно, ему надо заплатить всего лишь D_i йен, вместо суммы их цен.

Гарантируется, что никакой инструмент не появляется в списке скидок более одного раза и $\max(Y_i, Y_j) < D_{i,j} < Y_i + Y_j$ для любой пары, где $D_{i,j}$ цена покупки со скидкой i -ого и j -ого инструментов при одновременной покупке.

Также гарантируется, что $1 \leq N \leq 100, 2 \leq M \leq 100, 1 \leq K_i \leq 100, 1 \leq P \leq M/2$ и $1 \leq X_i, Y_i \leq 1000$.

Формат выходных данных

Выведите максимальную прибыль, которую может получить Такеши.

Задача D. План эвакуации

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

В городе есть муниципальные здания и бомбоубежища, которые были специально построены для эвакуации служащих в случае ядерной войны. Каждое бомбоубежище имеет ограниченную вместительность по количеству людей, которые могут в нем находиться. В идеале все работники из одного муниципального здания должны были бы бежать к ближайшему бомбоубежищу. Однако, в таком случае, некоторые бомбоубежища могли бы переполниться, в то время как остальные остались бы наполовину пустыми.

Чтобы разрешить эту проблему Городской Совет разработал специальный план эвакуации. Вместо того, чтобы каждому служащему индивидуально приписать, в какое бомбоубежище он должен бежать, для каждого муниципального здания определили, сколько служащих из него в какое бомбоубежище должны бежать. Задача индивидуального распределения была переложена на внутреннее управление муниципальных зданий.

План эвакуации учитывает количество служащих в каждом здании — каждый служащий должен быть учтен в плане и в каждое бомбоубежище может быть направлено количество служащих, не превосходящее вместимости бомбоубежища.

Городской Совет заявляет, что их план эвакуации оптимален в том смысле, что суммарное время эвакуации всех служащих города минимально.

Мэр города, находящийся в постоянной конфронтации с Городским Советом, не слишком то верит этому заявлению. Поэтому он нанял Вас в качестве независимого эксперта для проверки плана эвакуации. Ваша задача состоит в том, чтобы либо убедиться в оптимальности плана Городского Совета, либо доказать обратное, представив в качестве доказательства другой план эвакуации с меньшим суммарным временем для эвакуации всех служащих.

Карта города может быть представлена в виде квадратной сетки. Расположение муниципальных зданий и бомбоубежищ задается парой целых чисел, а время эвакуации из муниципального здания с координатами (X_i, Y_i) в бомбоубежище с координатами (P_j, Q_j) составляет $D_{ij} = |X_i - P_j| + |Y_i - Q_j| + 1$ минут.

Формат входных данных

Входной файл содержит описание карты города и плана эвакуации, предложенного Городским Советом. Первая строка входного файла содержит два целых числа N ($1 \leq N \leq 100$) и M ($1 \leq M \leq 100$), разделенных пробелом. N — число муниципальных зданий в городе (все они занумерованы числами от 1 до N), M — число бомбоубежищ (все они занумерованы числами от 1 до M).

Последующие N строк содержат описания муниципальных зданий. Каждая строка содержит целые числа X_i , Y_i и B_i , разделенные пробелами, где X_i , Y_i ($-1000 \leq X_i, Y_i \leq 1000$) — координаты здания, а B_i ($1 \leq B_i \leq 1000$) — число служащих в здании.

Описание бомбоубежищ содержится в последующих M строках. Каждая строка содержит целые числа P_j , Q_j и C_j , разделенные пробелами, где P_j , Q_j ($-1000 \leq P_j, Q_j \leq 1000$) — координаты бомбоубежища, а C_j ($1 \leq C_j \leq 1000$) — вместимость бомбоубежища.

В последующих N строках содержится описание плана эвакуации. Каждая строка представляет собой описание плана эвакуации для отдельного здания. План эвакуации из i -го здания состоит из M целых чисел E_{ij} , разделенных пробелами. E_{ij} ($0 \leq E_{ij} \leq 10000$) — количество служащих, которые должны эвакуироваться из i -го здания в j -е бомбоубежище.

Гарантируется, что план, заданный во входном файле, корректен.

Формат выходных данных

Если план эвакуации Городского Совета оптимален, то выведите одно слово **OPTIMAL**. В противном случае выведите на первой строке слово **SUBOPTIMAL**, а в последующих N строках выведите

Ваш план эвакуации (более оптимальный) в том же формате, что и во входном файле. Ваш план не обязан быть оптимальным, но должен быть лучше плана Городского Совета.

Примеры

стандартный ввод	стандартный вывод
3 4 -3 3 5 -2 -2 6 2 2 5 -1 1 3 1 1 4 -2 -2 7 0 -1 3 3 1 1 0 0 0 6 0 0 3 0 2	SUBOPTIMAL 3 0 1 1 0 0 6 0 0 4 0 1
3 4 -3 3 5 -2 -2 6 2 2 5 -1 1 3 1 1 4 -2 -2 7 0 -1 3 3 0 1 1 0 0 6 0 0 4 0 1	OPTIMAL

Задача Е. Мороженое

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

По дороге в школу Петя любит забегать в киоск и покупать себе мороженое. Однако при этом он часто опаздывает в школу. Неожиданно Петя понял — он просто ходит не по кратчайшему пути!

Помогите Пете победить опоздания. Город можно представить как n перекрестков, соединенных m улицами, про каждую улицу известна ее длина. Дом Пети находится на перекрестке a , школа — на перекрестке b , а киоск с мороженым — на перекрестке c . По пути в школу Петя никогда не проходит через один перекресток дважды.

Формат входных данных

Первая строка входного файла содержит числа n и m ($3 \leq n \leq 3 \cdot 10^4, 0 \leq m \leq 5 \cdot 10^4$). Вторая строка содержит три различных числа — a , b и c . Следующие m строк содержат по три целых числа x_i , y_i и l_i — номера перекрестков, соединенных улицей и ее длину (длина — целое неотрицательное число, которое не превышает 10^4).

Формат выходных данных

Если путь из дома Пети до школы, проходящие через перекресток с мороженым и не проходящий по одному перекрестку два раза, существует, выведи на первой строке выходного файла два целых числа k и l — количество улиц, которые проходит Петя в оптимальном пути и длину пути. На второй строке выведите номера перекрестков в том порядке, в котором их посещает Петя.

В противном случае выведите -1 на первой строке выходного файла.

Пример

стандартный ввод	стандартный вывод
5 6 1 5 3 1 2 1 2 3 4 2 4 1 3 4 2 1 5 1 4 5 2	4 9 1 2 3 4 5

Задача F. Поток в двудольном графе

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Дан двудольный граф, исток и сток. Каждая доля состоит из N вершин. Из истока в i -ю вершину левой доли ведёт ребро пропускной способности a_i . Из i -й вершины правой доли ведёт ребро в сток пропускной способности b_i . Также между некоторыми вершинами левой и правой доли есть двусторонние ребра бесконечной пропускной способности.

Вам требуется найти величину максимального потока из истока в сток.

Формат входных данных

В первой строке заданы два числа N и M ($1 \leq N \leq 10^4$, $0 \leq M \leq 10^5$) — число вершин в каждой доле и число рёбер соответственно.

Во второй строке записаны N чисел a_i ($1 \leq a_i \leq 10^4$).

Во третьей строке записаны N чисел b_i ($1 \leq b_i \leq 10^4$).

В каждой из последующих M строк записаны пары чисел u, v ($1 \leq u, v \leq n$) означающие наличие ребра между вершиной u левой доли и вершиной v правой доли.

Формат выходных данных

Выведите одно число — величину максимального потока.

Пример

стандартный ввод	стандартный вывод
3 4 3 2 1 5 4 4 1 1 1 2 2 3 3 3	6